Distributed Anomaly Detection Using Satellite Data From Multiple Modalities

Kanishka Bhaduri[1] Kamalika Das[2] Petr Votava[3]

[1]MCT Inc.
NASA Ames Research Center, Moffett Field CA 94035

[2]SGT Inc.
NASA Ames Research Center, Moffett Field CA 94035

[3]CSU Monterey Bay
NASA Ames Research Center, Moffett Field CA 94035

CIDU 2010

Roadmap

Introduction

- Massive volumes of earth science data collected and generated by growing number of satellites, in-situ sensors and increasingly complex ecosystem and climate models
- Identification of anomalies within the ecosystems
  - wildfires, droughts, floods, insect/pest damage, wind damage, logging
- Datasets stored at geographically different locations
  - NASA's Distributed Active Archive Centers (DAAC) stores Earth science data at 12 locations
- Scalable algorithms needed to co-analyze these peta-byte scale distributed data sources

Contribution

- Scalable algorithm for distributed anomaly detection on vertically partitioned data
- Communication required less than 1% of that required for centralization, yet 99% accurate compared to a centralized algorithm
- Capable of detecting significant outliers missed by using only a subset of features

Background: ν 1-class SVM

- Semi-supervised learning method for drawing a separating hyperplane that separates "+" from "-" or "good" from "bad"
- ν 1-class SVM draws separating hyperplane with ν % of data on one side
    - Design parameter ν: maximal rate of outliers in training set

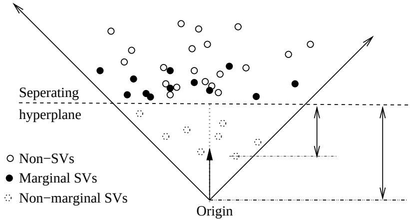

Figure: ν 1-class SVM

Background: ν 1-class SVM ...contd.

- Non-linear hyperplane in input space formed by kernel:

$$k(\overrightarrow{x_i}, \overrightarrow{x_j}) = \exp\left(\frac{-\left\|\overrightarrow{x_i} - \overrightarrow{x_j}\right\|^2}{2\sigma^2}\right)$$

Optimization problem

$$\begin{aligned} \text{minimize} \quad & Q = \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j k(\overrightarrow{x_i}, \overrightarrow{x_j}) + \rho\left(\nu m - \sum_i \alpha_i\right) \\ \text{subject to} \quad & 0 \leq \alpha_i \leq 1, \quad \nu \in [0,1] \end{aligned}$$

Background: ν 1-class SVM ...contd.

- Hyperplane defined by support vectors — points in the dataset which have $0 < \alpha_i \leq 1$
- For test point $\overrightarrow{x_t}$

$$f(\overrightarrow{x_t}) = \sum_{i \in SV} \alpha_i k(\overrightarrow{x_i}, \overrightarrow{x_t}) - \rho$$

- $\overrightarrow{x_t}$ outlier if $f(\overrightarrow{x_t}) < 0$

Distributed outlier detection: overview

- $P_0, ..., P_p$: nodes
- $D_i = \left[\overrightarrow{x_1^{(i)}} \quad \dots \quad \overrightarrow{x_m^{(i)}} \right]^T$: data at node P_i
  - Same m rows at each node
  - n_i features at node P_i

Distributed outlier detection: overview

- $P_0, ..., P_p$: nodes
- $D_i = \begin{bmatrix} \overrightarrow{x_1^{(i)}} & \dots & \overrightarrow{x_m^{(i)}} \end{bmatrix}^T$: data at node P_i
    - Same m rows at each node
    - n_i features at node P_i

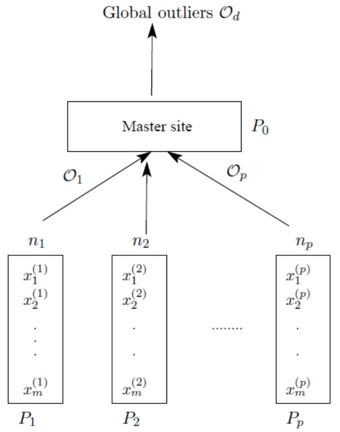

Figure: Computing model

Distributed outlier detection: local pruning rule

Pruning rule

An observation $\overrightarrow{x} \in D$ is a global outlier with respect to all the features if it is an outlier with respect to at least one (or a subset) of the features

Distributed outlier detection: local pruning rule

Pruning rule

An observation $\overrightarrow{x} \in D$ is a global outlier with respect to all the features if it is an outlier with respect to at least one (or a subset) of the features

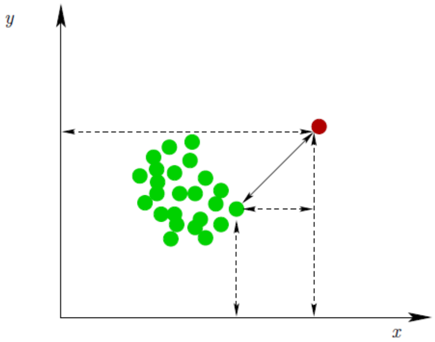

Figure: Local pruning rule

Local distributed outlier detection at P_i

Input D_i, sample size T_s, ν

Output outlier set $\mathcal{O}_i$

Process
- Get T_s samples from D_i for training SVM
- Test remaining points in D_i
- Send to P_0 those points in D_i whose anomaly score < 0

Global distributed outlier detection at P_0

Input $\mathcal{O}_1, \ldots, \mathcal{O}_p, T_s, \nu$

Output global outliers $\mathcal{O}_d$

Process

- Fetch T_s samples from each site for training global SVM at P_0
- Test all points in $\bigcup_i \mathcal{O}_i$
- Set all points with anomaly score < 0 as global outliers $\mathcal{O}_d$

California MODIS dataset

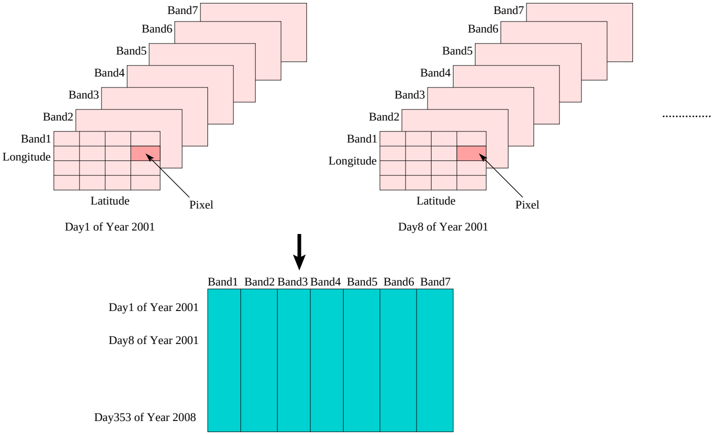

Figure: Preprocessing the CA MODIS dataset

Algorithm performance: accuracy

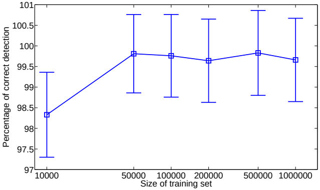

Figure: Accuracy on CA MODIS dataset

Algorithm performance: accuracy

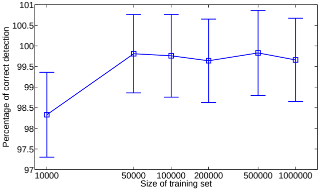

Figure: Accuracy on CA MODIS dataset

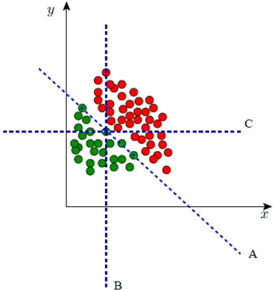

Figure: Correctness of distributed algorithm

Algorithm performance: running time

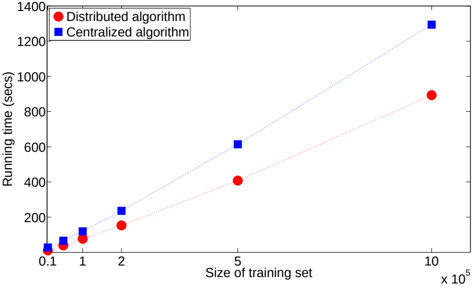

Figure: Running time CA MODIS dataset

Algorithm performance: running time

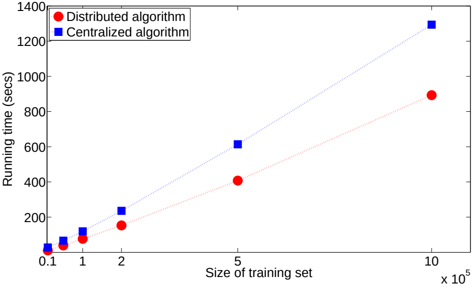

Figure: Running time CA MODIS dataset

Centralized: $O\left(m\left(\sum_{i=1}^{p} n_i\right)^2\right)$

Distributed: $O(mn_i^2)$

Algorithm performance: message complexity

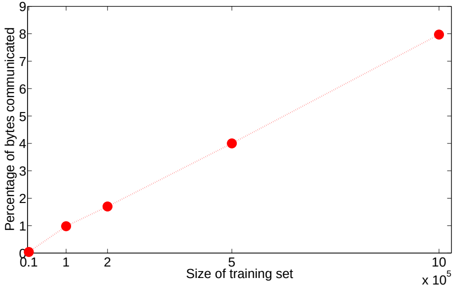

Figure: Message complexity on CA MODIS dataset

Algorithm performance: message complexity

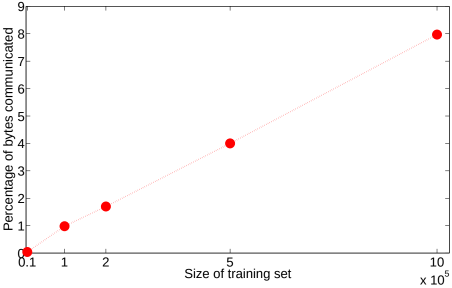

Figure: Message complexity on CA MODIS dataset

Centralized: $m \times \sum_{i=1}^{p} n_i$

Distributed: $\sum_{i=1}^{p} |\mathcal{O}_i| \times n_i + T_s \sum_{i=1}^{p} n_i$

Outliers on CA MODIS dataset

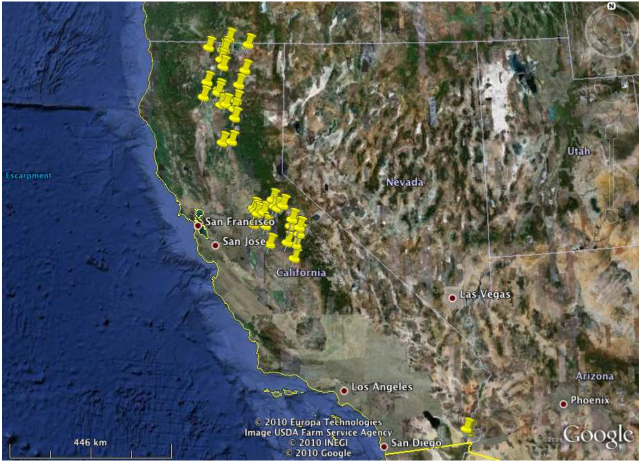

Figure: Top 50 unique outliers detected by the distributed algorithm

Summary

- Developed a distributed algorithm capable of detecting outliers from distributed data where each site has a subset of the global set of features
- Pruning rule achieves 99% accuracy with only 1% of the communication cost needed for centralization
- Future work is to extend this method for monitoring a data stream for outliers